Scientific Journal of Informatics
Vol. 11, No. 1, Feb 2024
p-ISSN 2407-7658 http://journal.unnes.ac.id/nju/index.php/sji e-ISSN 2460-0040

The Comparison of K-Nearest Neighbors and Random Forest Algorithm to Recognize Indonesian Sign Language in a Real-Time

Aaqila Dhiyaanisafa Goenawan[1*], Sri Hartati[2]
[1,2]Department of Computer Science, STMIK ESQ, Indonesia

Abstract.
Purpose: Comparing 2 models or prototype programs which can recognize Indonesian Sign Language System or *Sistem Isyarat Bahasa Indonesia* (SIBI) fonts from *hand gesture* and translate it's into writing Messages in real-time.
Methods: After selecting datasets and reprocessed by the researcher into 1 dataset, which are a combination of several sign image datasets of the SIBI letters images available on the Kaggle website, the dataset is converted into landmarks. The landmarks are divided into 26 sign classes and preprocessed to a total of 19,826 rows of data, and then divided into 67% training data and 33% test data. Next, both K-NN and Random Forest algorithm are implemented into different program and get tested into 2 different tests, model evaluation and real-time. At the end, the result is compared to see the increase of accuracy level of both K-Nearest Neighbors (K-NN) and Random Forest algorithm.
Result: The constructed and trained model is then evaluated and the results of Precision, Recall, Accuracy, and F1-Score are 99.88% using the Random Forest algorithm. The results of real-time program testing with the K-Nearest Neighbors algorithm get higher results, where the average accuracy value reaches 99%.
Novelty: From the result shows that the model built with the Random Forest algorithm is superior, but the K-Nearest Neighbors algorithm is better in real-time testing. Therefore, image data and its diversity should be increased, in order to improve recognition accuracy. The program could be enhanced by adding a function where the program can recognize hand gesture, not only one or two hands but also can recognize a hand gesture with movements so the program can recognize static and dynamic letter (required hands movement).

Keywords: SIBI application, Hand gesture recognition, K-NN, Random forest, Accuracy
Received November 2023 / **Revised** February 2024 / **Accepted** February 2024

INTRODUCTION

Sign language is a language that prioritizes communication using body language and lip movements. Sign language is generally used by the deaf community, combining movements of the hands, arms, body and facial expressions to express what they want to say. In 2018, Adi Kusumo Baroto, a researcher in Indonesian Sign Language, revealed to Berita Tempo that there are two sign languages used by the deaf community in Indonesia, namely Bahasa Isyarat Indonesia (BISINDO) and the Sistem Isyarat Bahasa Indonesia (SIBI). However, as the references and records regarding the BISINDO sign language were very minimal, the Indonesian government then created a sign language which has been standardized for use in various schools or special institutions, namely the Sistem Isyarat Bahasa Indonesia (SIBI)[28][35][36].

Like any languages in general, sign languages also have dictionaries or translators that have been agreed upon by both the government and the deaf community. The dictionary of the Sistem Isyarat Bahasa Indonesia (SIBI), contains movements and forms of hand signals that represent letters and existing terms. With our current technology, SIBI sign language dictionary are no longer has to be in a written record form, but in a form of a program that can recognize hand gestures from the sign language. In other words, it's very possible if we made a program which can be recognize hand gestures movements of people--even in a real time moment, by using the data-training which has been embedded into the Machine Learning as knowledge based for the Sign Language recognition program.

*Corresponding author.
Email addresses: a.dhiyaanisafa.g@students.esqbs.ac.id (Goenawan), srihartati@esqbs.ac.id (Hartati)
DOI: 10.15294/sji.v11i1.48475

The methodology used in research on hand gesture recognition can be applied by various algorithms, such as Deep Learning algorithm, namely Convolutional Neural Network (CNN), Support Vector Machine (SVM), K-Nearest Neighbors (K-NN), Random Forest, and other algorithms [5][10][21][23][24][29]. K-Nearest Neighbors (K-NN) algorithm is an algorithm that is often used on gesture recognition. In previous researches, according to Taunk et al. in [32], K-Nearest Neighbor (K-NN) algorithm is a machine learning algorithm that is non-parametric and a lazy learning algorithm [11][14][19][26][36], which means that the algorithm does not make assumptions about the distribution of the underlying data, it does not use training data to create a model, but only stores and remembers the training data, so it can be said to be lighter in terms of computing load and time [10][11][19][33][36]. Meanwhile, the Convolutional Neural Network (CNN) method was known less effective due to the heavy computational load or time [7][24][31].

Random Forest is an ensemble of numerous tree predictors-based approach in which N decision trees are independently built by bootstrapping the data. Put forward by Leo Breiman and Adele Culter, random forest, an integrated non-parametric learning machine with the purpose of sample training. The training speed is fast, the implementation of the algorithm is simple and effective, a classifier with high accuracy can be generated for varieties of data. Nevertheless, will be overfitting when dealing with some classification and regression problems with loud noise, the property weights produced by random forests on the data with different value properties are not credible [37][41].

This research was conducted to develop previous research on algorithm programs in sign language recognition of letters from hand movements. Some of these researches includes study was conducted by Damatraseta et al., which discussed real-time hand gesture recognition from videos using the Convolutional Neural Network (CNN) algorithm; research using the K-Nearest Neighbors (K-NN) algorithm by Alksasbeh et al [33]*;* and research using the Random Forest algorithm by Wiraswendro et al [34]. Specifically, this research was conducted to develop a program to hand gesture recognition of Indonesian Sign Language System (SIBI) sign language letters from hand movements directly (real-time) by applying two classification algorithms, namely the K-Nearest Neighbors and Random Forest algorithms, with the aim to get the best comparison of them especially in accuracy values, both in the model or the results terms. In addition, this research expected be a reference for creates communication media that is able to translate sign language letters from gestures into written form with the result to be a connector communication between deaf and non-deaf communities.

METHODS

The topics of this research studies include the SIBI sign language, hand gesture recognition, K-NN algorithm, and Random Forest algorithm. The data collected is in the form of a dataset sourced from the Kaggle site, which is sets of images of SIBI letters which will then be read to form coordinate points (hand landmarks).

Data Used

The dataset consists of 26 hand shape alphabet images from ‘A to Y’ letters of Indonesian Sign Language System (SIBI), except ‘J’ and ‘Z’ letters, as described in Figure 1.

Figure 1. SIBI Letters

Training and Sampling Dataset

The 26 alphabet letters are divided into 26 folders according to the name of the letter class folder. Each folder contains image data with name and type adjusted. The frequency every single letter of dataset distributed as 1880 number letter image data with the total of 48,880 dataset. The frequency of SIBI letters image shown in Table 1.

Table 1. Frequency of SIBI Letters Image

No	Dataset	Frequency
1	A	1880
2	B	1880
3	C	1880
…	…	1880
…	…	1880
26	Z	1880
Total		**48.880**

Dataset division was processed using the Scikit-learn library were 13,287 training data and 6,545 test data. The dataset converted into landmarks, and would be read to form coordinate points (hand landmarks). According to Mediapipe's official documentation, Mediapipe's hand landmark model detects 21 coordinates based on the knuckles of the detected hand, as in the following image Figure 2.

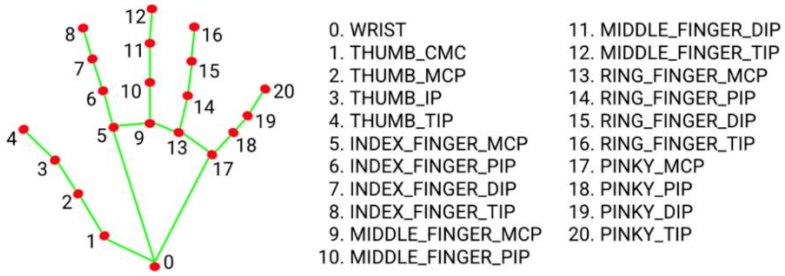


Figure 2. Mediapipe Hand Landmarks

Source:Hand landmarks detection guide of MediaPipe, Google for Developers, t.t.

The whole coordinates of image data will be extracted using Mediapipe Solutions, then will be manipulated into a dataset by The Pandas library, such as visualized in table form as below Figure 3.

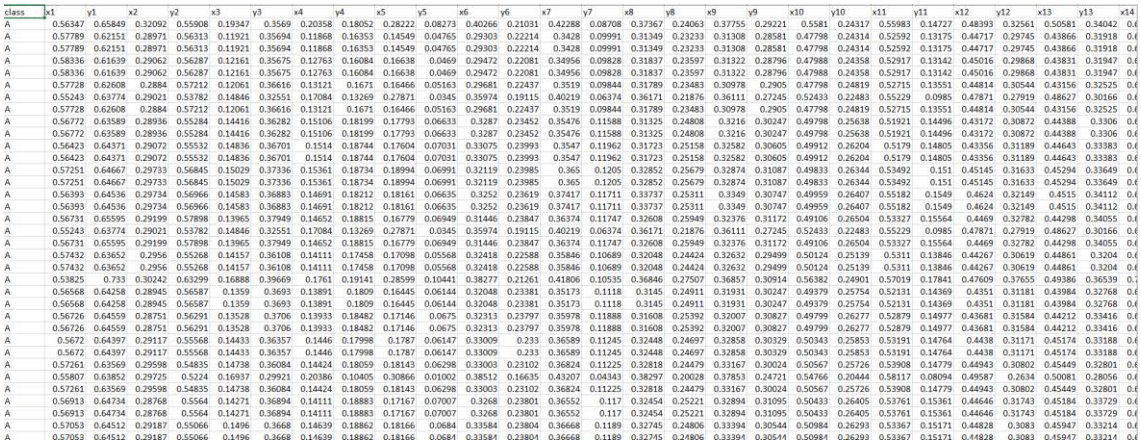
Out[21]:

	A	B	C	D	
0	./archive\asl_alphabet_train\A\A1.jpg	./archive\asl_alphabet_train\B\B1.jpg	./archive\asl_alphabet_train\C\C1.jpg	./archive\asl_alphabet_train\D\D1.jpg	./a
1	./archive\asl_alphabet_train\A\A10.jpg	./archive\asl_alphabet_train\B\B10.jpg	./archive\asl_alphabet_train\C\C10.jpg	./archive\asl_alphabet_train\D\D10.jpg	./ar
2	./archive\asl_alphabet_train\A\A100.jpg	./archive\asl_alphabet_train\B\B100.jpg	./archive\asl_alphabet_train\C\C100.jpg	./archive\asl_alphabet_train\D\D100.jpg	./arcl
3	./archive\asl_alphabet_train\A\A1000.jpg	./archive\asl_alphabet_train\B\B1000.jpg	./archive\asl_alphabet_train\C\C1000.jpg	./archive\asl_alphabet_train\D\D1000.jpg	./archi
4	./archive\asl_alphabet_train\A\A1001.jpg	./archive\asl_alphabet_train\B\B1001.jpg	./archive\asl_alphabet_train\C\C1001.jpg	./archive\asl_alphabet_train\D\D1001.jpg	./archi
...	...	...	...	...	
1875	./archive\asl_alphabet_train\A\A995.jpg	./archive\asl_alphabet_train\B\B995.jpg	./archive\asl_alphabet_train\C\C995.jpg	./archive\asl_alphabet_train\D\D995.jpg	./arcl
1876	./archive\asl_alphabet_train\A\A996.jpg	./archive\asl_alphabet_train\B\B996.jpg	./archive\asl_alphabet_train\C\C996.jpg	./archive\asl_alphabet_train\D\D996.jpg	./arcl
1877	./archive\asl_alphabet_train\A\A997.jpg	./archive\asl_alphabet_train\B\B997.jpg	./archive\asl_alphabet_train\C\C997.jpg	./archive\asl_alphabet_train\D\D997.jpg	./arcl
1878	./archive\asl_alphabet_train\A\A998.jpg	./archive\asl_alphabet_train\B\B998.jpg	./archive\asl_alphabet_train\C\C998.jpg	./archive\asl_alphabet_train\D\D998.jpg	./arcl
1879	./archive\asl_alphabet_train\A\A999.jpg	./archive\asl_alphabet_train\B\B999.jpg	./archive\asl_alphabet_train\C\C999.jpg	./archive\asl_alphabet_train\D\D999.jpg	./arcl

1880 rows × 24 columns

Figure 3. SIBI Letter Image Dataset

Furthermore, letter class selection is carried out, where each letter class is targeted at 1000 columns containing coordinates or landmark points from the dataset hand image, as shown in Figure 4.



class	x1	y1	x2	y2	x3	y3	x4	y4	x5	y5	x6	y6	x7	y7	x8	y8	x9	y9	x10	y10	x11	y11	x12	y12	x13	y13	x14
A	0.56347	0.65849	0.32092	0.55908	0.19347	0.3569	0.20358	0.18052	0.28222	0.08273	0.40266	0.21031	0.42288	0.08708	0.37367	0.24063	0.37755	0.29221	0.5581	0.24317	0.55983	0.14727	0.48393	0.32561	0.50581	0.34042	0.6
A	0.57789	0.62151	0.28971	0.56313	0.11921	0.35694	0.11868	0.16353	0.14549	0.04765	0.29303	0.22214	0.3428	0.09991	0.31349	0.23233	0.31308	0.28581	0.47798	0.24314	0.52592	0.13175	0.44717	0.29745	0.43866	0.31918	0.6
A	0.57789	0.62151	0.28971	0.56313	0.11921	0.35694	0.11868	0.16353	0.14549	0.04765	0.29303	0.22214	0.3428	0.09991	0.31349	0.23233	0.31308	0.28581	0.47798	0.24314	0.52592	0.13175	0.44717	0.29745	0.43866	0.31918	0.6
A	0.58336	0.61639	0.29062	0.56287	0.12161	0.35675	0.12763	0.16084	0.16638	0.0469	0.29472	0.22081	0.34956	0.09828	0.31837	0.23597	0.31322	0.28796	0.47988	0.24358	0.52917	0.13142	0.45016	0.29868	0.43831	0.31947	0.6
A	0.58336	0.61639	0.29062	0.56287	0.12161	0.35675	0.12763	0.16084	0.16638	0.0469	0.29472	0.22081	0.34956	0.09828	0.31837	0.23597	0.31322	0.28796	0.47988	0.24358	0.52917	0.13142	0.45016	0.29868	0.43831	0.31947	0.6
A	0.57728	0.62608	0.2884	0.57212	0.12061	0.36616	0.13121	0.1671	0.16466	0.05163	0.29681	0.22437	0.3519	0.09844	0.31789	0.23483	0.30978	0.2905	0.47798	0.24819	0.52715	0.13551	0.44814	0.30544	0.43156	0.32525	0.6
A	0.55243	0.63774	0.29021	0.53782	0.14846	0.32551	0.17084	0.13269	0.27871	0.0345	0.35974	0.19115	0.40219	0.06374	0.36171	0.21876	0.36111	0.27245	0.52433	0.22483	0.55229	0.0985	0.47871	0.27919	0.48627	0.30166	0.6
A	0.57728	0.62608	0.2884	0.57212	0.12061	0.36616	0.13121	0.1671	0.16466	0.05163	0.29681	0.22437	0.3519	0.09844	0.31789	0.23483	0.30978	0.2905	0.47798	0.24819	0.52715	0.13551	0.44814	0.30544	0.43156	0.32525	0.6
A	0.56772	0.63589	0.28936	0.55284	0.14416	0.36282	0.15106	0.18199	0.17793	0.06633	0.3287	0.23452	0.35476	0.11588	0.31325	0.24808	0.3216	0.30247	0.49798	0.25638	0.51921	0.14496	0.43172	0.30872	0.44388	0.3306	0.6
A	0.56772	0.63589	0.28936	0.55284	0.14416	0.36282	0.15106	0.18199	0.17793	0.06633	0.3287	0.23452	0.35476	0.11588	0.31325	0.24808	0.3216	0.30247	0.49798	0.25638	0.51921	0.14496	0.43172	0.30872	0.44388	0.3306	0.6
A	0.56423	0.64371	0.29072	0.55532	0.14836	0.36701	0.1514	0.18744	0.17604	0.07031	0.33075	0.23993	0.3547	0.11962	0.31723	0.25158	0.32582	0.30605	0.49912	0.26204	0.5179	0.14805	0.43356	0.31189	0.44643	0.33383	0.6
A	0.56423	0.64371	0.29072	0.55532	0.14836	0.36701	0.1514	0.18744	0.17604	0.07031	0.33075	0.23993	0.3547	0.11962	0.31723	0.25158	0.32582	0.30605	0.49912	0.26204	0.5179	0.14805	0.43356	0.31189	0.44643	0.33383	0.6
A	0.57251	0.64667	0.29733	0.56845	0.15029	0.37336	0.15361	0.18734	0.18994	0.06991	0.32119	0.23985	0.365	0.1205	0.32852	0.25679	0.32874	0.31087	0.49833	0.26344	0.53492	0.151	0.45145	0.31633	0.45294	0.33649	0.6
A	0.57251	0.64667	0.29733	0.56845	0.15029	0.37336	0.15361	0.18734	0.18994	0.06991	0.32119	0.23985	0.365	0.1205	0.32852	0.25679	0.32874	0.31087	0.49833	0.26344	0.53492	0.151	0.45145	0.31633	0.45294	0.33649	0.6
A	0.56393	0.64536	0.29734	0.56966	0.14583	0.36883	0.14691	0.18212	0.18161	0.06635	0.3252	0.23619	0.37417	0.11711	0.33737	0.25311	0.3349	0.30747	0.49959	0.26407	0.55182	0.1549	0.4624	0.32149	0.4515	0.34112	0.6
A	0.56393	0.64536	0.29734	0.56966	0.14583	0.36883	0.14691	0.18212	0.18161	0.06635	0.3252	0.23619	0.37417	0.11711	0.33737	0.25311	0.3349	0.30747	0.49959	0.26407	0.55182	0.1549	0.4624	0.32149	0.4515	0.34112	0.6
A	0.56731	0.65595	0.29199	0.57898	0.13965	0.37949	0.14652	0.18815	0.16779	0.06949	0.31446	0.23847	0.36374	0.11747	0.32608	0.25949	0.32376	0.31172	0.49106	0.26504	0.53327	0.15564	0.4469	0.32782	0.44298	0.34055	0.6
A	0.55243	0.63774	0.29021	0.53782	0.14846	0.32551	0.17084	0.13269	0.27871	0.0345	0.35974	0.19115	0.40219	0.06374	0.36171	0.21876	0.36111	0.27245	0.52433	0.22483	0.55229	0.0985	0.47871	0.27919	0.48627	0.30166	0.6
A	0.56731	0.65595	0.29199	0.57898	0.13965	0.37949	0.14652	0.18815	0.16779	0.06949	0.31446	0.23847	0.36374	0.11747	0.32608	0.25949	0.32376	0.31172	0.49106	0.26504	0.53327	0.15564	0.4469	0.32782	0.44298	0.34055	0.6
A	0.57432	0.63652	0.2956	0.55268	0.14157	0.36108	0.14111	0.17458	0.17098	0.05568	0.32418	0.22588	0.35846	0.10689	0.32048	0.24424	0.32632	0.29499	0.50124	0.25139	0.5311	0.13846	0.44267	0.30619	0.44861	0.3204	0.6
A	0.57432	0.63652	0.2956	0.55268	0.14157	0.36108	0.14111	0.17458	0.17098	0.05568	0.32418	0.22588	0.35846	0.10689	0.32048	0.24424	0.32632	0.29499	0.50124	0.25139	0.5311	0.13846	0.44267	0.30619	0.44861	0.3204	0.6
A	0.53825	0.733	0.30242	0.63299	0.16888	0.39669	0.1761	0.19141	0.28599	0.10441	0.38277	0.21261	0.41806	0.10535	0.36846	0.27507	0.36857	0.30914	0.56382	0.24901	0.57019	0.17841	0.47609	0.37655	0.49386	0.36539	0.7
A	0.56568	0.64258	0.28945	0.56587	0.1359	0.3693	0.13891	0.1809	0.16445	0.06144	0.32048	0.23381	0.35173	0.1118	0.3145	0.24911	0.31931	0.30247	0.49379	0.25754	0.52131	0.14369	0.4351	0.31181	0.43984	0.32768	0.6
A	0.56568	0.64258	0.28945	0.56587	0.1359	0.3693	0.13891	0.1809	0.16445	0.06144	0.32048	0.23381	0.35173	0.1118	0.3145	0.24911	0.31931	0.30247	0.49379	0.25754	0.52131	0.14369	0.4351	0.31181	0.43984	0.32768	0.6
A	0.56726	0.64559	0.28751	0.56291	0.13528	0.3706	0.13933	0.18482	0.17146	0.0675	0.32313	0.23797	0.35978	0.11888	0.31608	0.25392	0.32007	0.30827	0.49799	0.26277	0.52879	0.14977	0.43681	0.31584	0.44212	0.33416	0.6
A	0.56726	0.64559	0.28751	0.56291	0.13528	0.3706	0.13933	0.18482	0.17146	0.0675	0.32313	0.23797	0.35978	0.11888	0.31608	0.25392	0.32007	0.30827	0.49799	0.26277	0.52879	0.14977	0.43681	0.31584	0.44212	0.33416	0.6
A	0.5672	0.64397	0.29117	0.55568	0.14433	0.36357	0.1446	0.17998	0.1787	0.06147	0.33009	0.233	0.36589	0.11245	0.32448	0.24697	0.32858	0.30329	0.50343	0.25853	0.53191	0.14764	0.4438	0.31171	0.45174	0.33188	0.6
A	0.5672	0.64397	0.29117	0.55568	0.14433	0.36357	0.1446	0.17998	0.1787	0.06147	0.33009	0.233	0.36589	0.11245	0.32448	0.24697	0.32858	0.30329	0.50343	0.25853	0.53191	0.14764	0.4438	0.31171	0.45174	0.33188	0.6
A	0.57261	0.63569	0.29598	0.54835	0.14738	0.36084	0.14424	0.18059	0.18143	0.06298	0.33003	0.23102	0.36824	0.11225	0.32818	0.24479	0.33167	0.30024	0.50567	0.25726	0.53908	0.14779	0.44943	0.30802	0.45449	0.32801	0.6
A	0.57261	0.63569	0.29598	0.54835	0.14738	0.36084	0.14424	0.18059	0.18143	0.06298	0.33003	0.23102	0.36824	0.11225	0.32818	0.24479	0.33167	0.30024	0.50567	0.25726	0.53908	0.14779	0.44943	0.30802	0.45449	0.32801	0.6
A	0.55807	0.63852	0.29725	0.5224	0.16937	0.29921	0.20386	0.10405	0.30866	0.01002	0.38512	0.16635	0.43207	0.04343	0.38297	0.20028	0.37853	0.24721	0.54766	0.20444	0.58117	0.08094	0.49587	0.2634	0.50081	0.28056	0.6
A	0.56913	0.64734	0.28768	0.5564	0.14271	0.36894	0.14111	0.18883	0.17167	0.07007	0.3268	0.23801	0.36552	0.117	0.32454	0.25221	0.32894	0.31095	0.50433	0.26405	0.53761	0.15361	0.44646	0.31743	0.45184	0.33729	0.6
A	0.56913	0.64734	0.28768	0.5564	0.14271	0.36894	0.14111	0.18883	0.17167	0.07007	0.3268	0.23801	0.36552	0.117	0.32454	0.25221	0.32894	0.31095	0.50433	0.26405	0.53761	0.15361	0.44646	0.31743	0.45184	0.33729	0.6
A	0.57053	0.64512	0.29187	0.55066	0.1496	0.3668	0.14639	0.18862	0.18166	0.0684	0.33584	0.23804	0.36668	0.1189	0.32745	0.24806	0.33394	0.30544	0.50984	0.26293	0.53367	0.15171	0.44828	0.3083	0.45947	0.33214	0.6
A	0.57053	0.64512	0.29187	0.55066	0.1496	0.3668	0.14639	0.18862	0.18166	0.0684	0.33584	0.23804	0.36668	0.1189	0.32745	0.24806	0.33394	0.30544	0.50984	0.26293	0.53367	0.15171	0.44828	0.3083	0.45947	0.33214	0.6

Figure 4. Landmarks of SIBI Letter by Mediapipe Solutions

The next process, landmarks are divided into 26 sign classes and preprocessed to a total of 19,826 rows of data, and then divided into 67% training data and 33% test data. The dataset will be processes and test using the model which is built using K-Nearest Neighbors, Random Forest algorithm and the Python programming language version 3.9 with *Jupyter Notebook*, *Mediapipe* and *scikit-learn libraries.* To visualization flow for data testing the System of Indonesian Sign Language (SIBI) is described in below Diagram Flow Figure 5.

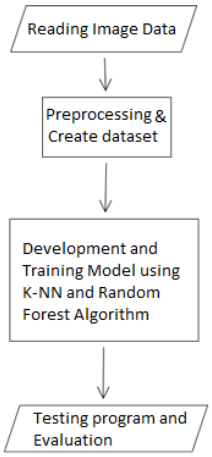


Figure 5. Data Testing Flow

The testing process is divided into two processes, that is model evaluation testing and real-time program testing. The results of the testing processes are carried out by comparing the results of the precision, F1 Score, and recall produced during the evaluation test, as well as the accuracy of real-time hand gesture recognition of SIBI letter.

Comparative of K-Nearest Neighbors and Random Forest Algorithms

According to [11][12][18][19][33][36], the way K-NN algorithm works is by determining the number of neighbors (*k*) which will be used as an estimate for class determination. The formula for calculating similarity weights with the K-NN algorithm is to use the Euclidean distance (*D*) formula from training data (x) and test data (*y*), which is written in the following equation:

$$D_{xy} = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (1)$$

The Random Forest algorithm is a development of the decision tree algorithm, which is a collection of classification tree structures as following:

$$\{h(x,k\,),k\ =\ 1,\dots\} \quad (2)$$

The k values are random vectors that are distributed independently and identically, and each tree produces one of the most popular results given the input x. Quoting from Breiman, the classification of the Random Forest Algorithm is determined based on the voting results of the most popular results. The tree collection was obtained through the bootstrap aggregating (bagging) process [2][4][7][8][38]. This algorithm works by combining n decision trees to build a random forest. Then, this process is continued with each tree to produce predictions, which will then be carried out by a voting process using the mode (the value that appears most frequently) to select the final prediction result [22][24][31][39].

To initialize the RandomForestClassifier class, the decision tree is determined randomly with the n_estimators command and controlled using the random_state parameter as much as 1234. Subsequently, after a hyperparameter search completed, then based on the best results of hypertuning model process, it is determined that the number of parameters such as decision trees, max_depth, and max_features is 100 trees.

RESULTS AND DISCUSSIONS

Real-time Testing

Program testing was carried out in real-time using a Dell G3 3500 Laptop webcam with an RGB HD Camera type and a camera resolution of 1280 x 720 (HD) at 30 fps. In this testing process, letter recognition was measured based on the distance between the hand and the webcam. To start the hand detection and recognition process, the program captured landmark coordinates as view on the Figure 6, then the model will be using them as data 'memorized'.

Figure 6. Real-time SIBI Letter Recognition

Result

As previously mentioned, the aim of research using two the two algorithms namely K-Nearest Neighbors and Random Forest, is to see a comparison of them, especially in terms of Accuracy, Precision and Recall evaluation tests. Referring to the model we created in this research, using a dataset of 24 SIBI letter classes which is further divided into 13,322 training data and 6562 test data, calculations *measured* using the Multi Class Confusion Matrix produce data interpretation [39][40] *as* presented in Table 2 and below Figure 7.

Table 2. Comparison of the calculations results between *KNN and RF Algorithm* in Accuracy, Recall, Precision, and F1-Score

Measurement	KNN		Random Forest	
	Prev. Research [10]	Development	Prev. Research [11]	Development
Accuracy	97.12%	99.60%	98.6%	99.88%
Recall	97.09%	99.60%	.98.66%	99.88%
Precision	(not specified)	99.62%	98.66%	99.88%
F1-Score	(not specified)	99.60%	(not specified)	99.88%

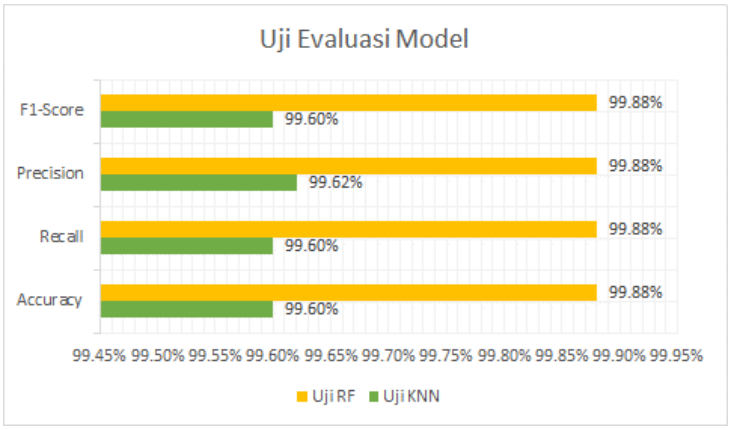


Figure 7. Chart Evaluation Test Comparison of Model KNN and RF Algorithms

Based on Table 1 and Figure 7 above, it can be concluded that the model applied with the Random Forest algorithm has a higher value than the model applied with the K-Nearest Neighbors algorithm, which is model using the K-Nearest Neighbors (K-NN) algorithm resulted a score of 99.60% for its F1-Score, Recall and Accuracy values; meanwhile, the model using Random Forest algorithm resulted a score of 99.88% for its F1-Score, Precision, Recall and Accuracy values. However, the level of accuracy obtained in this study was able to exceed the level of accuracy of previous research.

Discussion

The model of this research can only recognize SIBI letters through landmark patterns of static hand sign language image, and do not recognize from videos with dynamic movements. So that, it can only recognize 24 letter classes from 26 hand sign language A to Z of the alphabet letters in real time, except for the J and Z letters, because of these two letters require movement to demonstrate. The development of this research also can only compare the result of *accuracy; precision; F1-Score; and recall* by applying each K-Nearest Neighbors (K-NN) algorithm and the Random Forest algorithm to the model then testing separately, so it has not yet reached the experimental stage of combining the two algorithms.

CONCLUSION

Main objective of this research is to compare two algorithms, are K-Nearest Neighbors and Random Forest in sign language recognition of SIBI letters from real time hand movements, as well as increasing results the accuracy of research previously, especially in terms of Accuracy; Precision; F1-Score; and Recall. In this our model, from 26 alphabet letters, only uses the 24 letter SIBI image data dataset (without the letters 'J' and 'Z') with the total of 48,880 datasets, then processed to landmarks coordinates with a total of 19,826 rows landmarks data. Then, the landmarks data divided into 13,322 (67%) training data and 6562 (33%) test data. The model with the application of the K-Nearest Neighbors (K-NN) algorithm resulted in a score of 99.60% with an increase in accuracy of 2.48% from previous research was a score as 97.12%. Meanwhile, the model using Random Forest algorithm resulted in a score of 99.88%, an increase from 97.12% or in other words, it resulted increase in accuracy of 1.28% from previous research.

REFERENCES

[1] S, A. Potluri, S. M. George, G. R and A. S, "Indian Sign Language Recognition Using Random Forest Classifier," 2021 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), Bangalore, India, 2021, pp. 1-6, doi: 10.1109/CONECCT52877.2021.9622672.

[2] Su, R.; Chen, X.; Cao, S.; Zhang, X. "Random Forest-Based Recognition of Isolated Sign Language Subwords Using Data from Accelerometers and Surface Electromyographic Sensors". *Sensors* **2016**, *16*, 100. https://doi.org/10.3390/s16010100.

[3] Kenshimov, Chingiz, et al. "Sign Language Dactyl Recognition Based on Machine Learning Algorithms." *Eastern-European Journal of Enterprise Technologies*, vol. 4, no. 2, 2021, pp. 58-72, doi:10.15587/1729-4061.2021.239253.

[4] Yugam Bajaj and Puru Malhotra "American Sign Language Identification Using Hand Trackpoint Analysis" Springer, arxiv:2010.10590

[5] Mahalakshmi V "Sign Language Training Tool Using Machine Learning Techniques" International Journal of Research Publication and Reviews, Vol 4, no 6, pp 3488-3494 June 2023
[6] Sunanda Das, et al "A hybrid approach for Bangla sign language recognition using deep transfer learning model with random forest classifier" Elsevier: Expert Systems with Applications Volume 213, Part B, 1 March 2023, 118914. https://doi.org/10.1016/j.eswa.2022.118914
[7] Radha S. Shirbhate, et al "Sign language Recognition Using Machine Learning Algorithm" International Research Journal of Engineering and Technology (IRJET) Volume: 07 Issue: 03 Mar 2020
[8] Rasha Amer Kadhim and Muntadher Khamees "A Real-Time American Sign Language Recognition System using Convolutional Neural Network for Real Datasets" TEM Journal. Volume 9, Issue 3, Pages 937-943, ISSN 2217-8309, DOI: 10.18421/TEM93-14, August 2020
[9] Hisham, B., Hamouda, A. "Supervised learning classifiers for Arabic gestures recognition using Kinect V2". *SN Appl. Sci.* **1**, 768 (2019). https://doi.org/10.1007/s42452-019-0771-2
[10] Jitendra Jaiswal "A Comparative Analysis On Sign Language Prediction Using Machine Learning Algorithms" 2020, https://api.semanticscholar.org/CorpusID:219616308
[11] Fitri Utaminingrum, et al "Alphabet Sign Language Recognition Using K-Nearest Neighbor Optimization". https://www.semanticscholar.org/ DOI:10.17706/jcp.14.1.63-70, Corpus ID: 59616174
[12] Arsheldy Alvin, et al "Hand Gesture Detection for American Sign Language using K-Nearest Neighbor with Mediapipe", 2021, https://api.semanticscholar.org/CorpusID:247358722
[13] Neeraj Kumar Pandey, et al "An Improved Sign Language Translation approach using KNN in Deep Learning Environment" https://api.semanticscholar.org/DOI:10.1109/ICDT57929.2023.10150934, Corpus ID: 259217415
[14] Madhuri Sharma, et al "Indian Sign Language Recognition Using Neural Networks and K-NN Classifiers" ARPN Journal of Engineering and Applied Sciences VOL. 9, NO. 8, AUGUST 2014 ISSN 1819-6608
[15] Dewinta and Yaya Heriyadi, "American Sign Language-Based Finger-spelling Recognition using k-Nearest Neighbours Classifier" Conference: The 3rd International Conference on Information and Communication Technology, Bali, Indonesia, May 2015, DOI:10.1109/ICoICT.2015.7231481. https://www.researchgate.net/publication/279198249_American_Sign_Language-Based_Finger-spelling_Recognition_using_k-Nearest_Neighbours_Classifier
[16] M. Rajanishree, N. Nadeem Ahmed, Y. Panchani, S. Aravindan and V. Jadhav, "Sign Language Conversion to Speech with the Application of KNN Algorithm," *2022 Sixth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, Dharan, Nepal, 2022, pp. 886-890, doi: 10.1109/I-SMAC55078.2022.9987421.
[17] N. B. Linsangan, J. V. G. Calites, J. T. L. Reyes, G. C. D. Sioson, R. V. Pellegrino and I. C. Juanatas, "Filipino Sign Language to Text Converter using K-Nearest Neighbor Algorithm," *2022 IEEE 14th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, Boracay Island, Philippines, 2022, pp. 1-6, doi: 10.1109/HNICEM57413.2022.10109512.
[18] S. Sharma, R. Sreemathy, M. Turuk, J. Jagdale and S. Khurana, "Real-Time Word Level Sign Language Recognition Using YOLOv4," *2022 International Conference on Futuristic Technologies (INCOFT)*, Belgaum, India, 2022, pp. 1-7, doi: 10.1109/INCOFT55651.2022.10094530.
[19] Malek Zakarya Alksasbeh, et al "Smart hand gestures recognition using K-NN based algorithm for video annotation purposes" The Indonesian Journal of Electrical Engineering and Computer Science (IJEECS), p-ISSN: 2502-4752, e-ISSN: 2502-4760, DOI: http://doi.org/10.11591/ijeecs.v21.i1.pp242-252
[20] Fifin Ayu Mufarroha and Fitri Utaminingrum "The Greatest Points in Hand Gesture Recognition for American Sign Language" International Journal of Intelligent Engineering and Systems, 2017
[21] I.A. Adeyanju , et al "Machine learning methods for sign language recognition: A critical review and analysis" Intelligent Systems with Applications Volume 12, November 2021, 200056. https://doi.org/10.1016/j.iswa.2021.200056
[22] Lester Wong Sze Ee, et al "Real-Time Sign Language Learning System" Journal of Physics: Conference Series 1712 (2020) 012011 IOP Publishing doi:10.1088/1742-6596/1712/1/012011
[23] Ahmed Sultan, et al "Sign language identification and recognition: A comparative study" journal Open Computer Science. https://doi.org/10.1515/comp-2022-0240
[24] Rupesh Kumar, et al "A Comparative Analysis of Techniques and Algorithms for Recognising Sign Language" Arxiv: 2305.13941
[25] Hany A. AbdElghfar, Abdelmoty M. Ahmed, Ali A. Alani, Hammam M. AbdElaal, Belgacem Bouallegue, Mahmoud M. Khattab, Gamal Tharwat, Hassan A. Youness, "A Model for Qur'anic Sign Language Recognition

Based on Deep Learning Algorithms", *Journal of Sensors*, vol. 2023, Article ID 9926245, 13 pages, 2023. https://doi.org/10.1155/2023/9926245

[26] Zafar Ahmed Ansari and Gaurav Harit "Nearest neighbour classification of Indian sign language gestures using kinect camera" Sadhana Vol. 41, No. 2, February 201, pp. 161–182

[27] Zahid H, Rashid M, Syed SA, Ullah R, Asif M, Khan M, Abdul Mujeeb A, Haider Khan A. 2022. A computer vision-based system for recognition and classification of Urdu sign language dataset. PeerJ Comput. Sci. 8:e1174 http://doi.org/10.7717/peerjcs.1174

[28] Nurhadi, "Mengenal Bisindo dan Sibi, 2 Bahasa Isyarat yang Digunakan di Indonesia - Difabel Tempo.co," Tempo.co. Accessed: Oct. 06, 2022. [Online]. Available: https://difabel.tempo.co/read/1624137/mengenal-bisindo-dan-sibi-2-bahasa-isyarat-yang-digunakan-di-indonesia.

[29] N. Aziz and A. Kurniawardhani, "The Development of Hand Gestures Recognition Research: A Review," *International Journal of Artificial Intelligence Research*, vol. 6, no. 1, Jun. 2021, doi: 10.29099/ijair.v6i1.236.

[30] K. Taunk, S. De, S. Verma, and A. Swetapadma, *A Brief Review of Nearest Neighbor Algorithm for Learning and Classification*. IEEE, 2019.

[31] Dertat, "Applied Deep Learning - Part 4: Convolutional Neural Networks," Medium. Accessed: Jan. 13, 2023. [Online]. Available: https://towardsdatascience.com/applied-deep-learning-part-4-convolutional-neural-networks-584bc134c1e2.

[32] F. Damatraseta, R. Novariany, and M. A. Ridhani, "Real-time BISINDO Hand Gesture Detection and Recognition with Deep Learning CNN," *Jurnal Informatika Kesatuan*, vol. 1, no. 1, pp. 71–80, Jul. 2021, doi: 10.37641/jikes.v1i1.774.

[33] M. Z. Alksasbeh *et al.*, "Smart hand gestures recognition using K-NN based algorithm for video annotation purposes," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 21, no. 1, pp. 242–252, Jan. 2021, doi: 10.11591/ijeecs.v21.i1.pp242-252.

[34] P. E. Wiraswendro and H. Soetanto, "Penerapan Algoritma Random Forest Classifier Pada Sistem Deteksi Simbol Sistem Isyarat Bahasa Indonesia (SIBI)," 2022. [Online]. Available: https://pmpk.kemdikbud.go.id/sibi/

[35] "Kamus SIBI." Accessed: Jun. 04, 2023. [Online]. Available: https://pmpk.kemdikbud.go.id/sibi/kosakata

[36] L. Afifah, "Algoritma K-Nearest Neighbor (KNN) untuk Klasifikasi - IlmudataPy." Accessed: Jan. 14, 2023. [Online]. Available: https://ilmudatapy.com/algoritma-k-nearest-neighbor-knn-untuk-klasifikasi/

[37] L. Breiman, "Random Forests," 2001. "What is Random Forest? | IBM," IBM. Accessed: May 22, 2023. [Online]. Available: https://www.ibm.com/topics/random-forest.

[38] Tharwat, "Classification assessment methods," *Applied Computing and Informatics*, vol. 17, no. 1, pp. 168–192, 2018, doi: 10.1016/j.aci.2018.08.003.

[39] Markoulidakis, G. Kopsiaftis, I. Rallis, and I. Georgoulas, "Multi-Class Confusion Matrix Reduction method and its application on Net Promoter Score classification problem," in *ACM International Conference Proceeding Series*, Association for Computing Machinery, Jun. 2021, pp. 412–419. doi: 10.1145/3453892.3461323.

[40] Liao, Xianghua & Zheng, Jiaxuan & Huang, Chengli & Huang, Guoru. (2018). Approach for Evaluating LID Measure Layout Scenarios Based on Random Forest: Case of Guangzhou—China. Water (Switzerland). 10. 10.3390/w10070894.